Carnegie Mellon
Software Engineering Institute

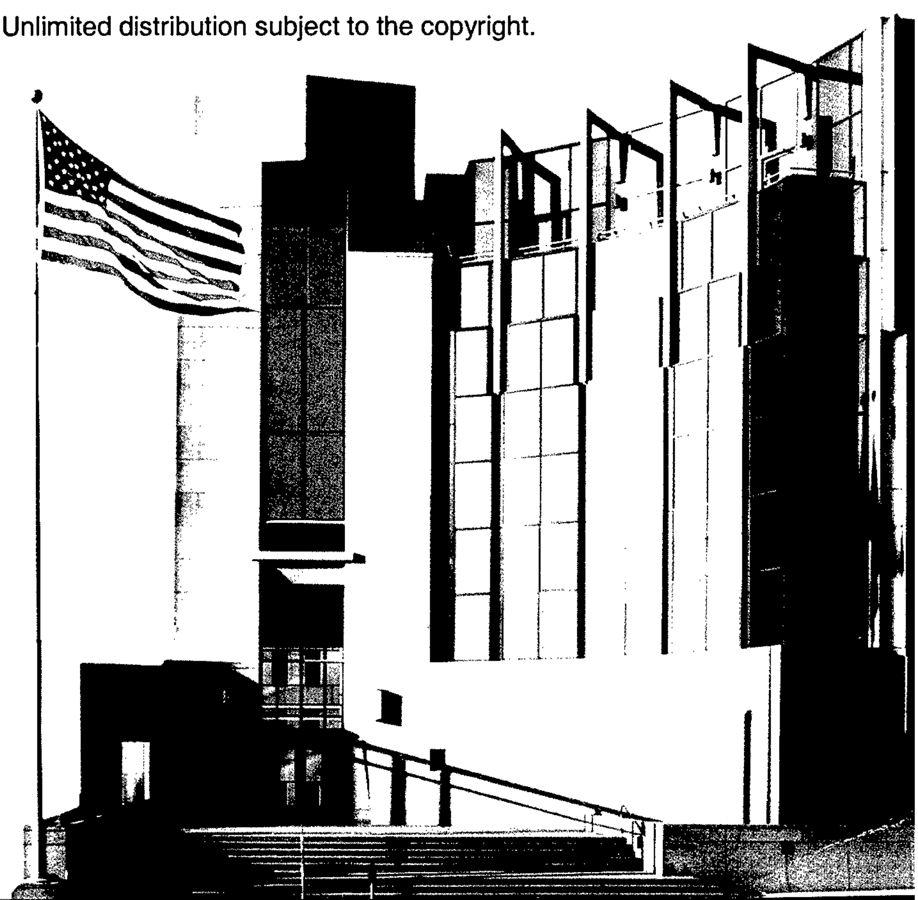
Using the SEI Architecture Tradeoff Analysis Method to Evaluate WIN-T: A Case Study

Paul Clements, Software Engineering Institute
John Bergey, Software Engineering Institute
Dave Mason, U.S. Army's Communications and Electronics Command (CECOM)

September 2005

Software Architecture Technology Initiative

20060515153



Technical Note
CMU/SEI-2005-TN-027

Using the SEI Architecture Tradeoff Analysis Method to Evaluate WIN-T: A Case Study

Paul Clements, Software Engineering Institute
John Bergey, Software Engineering Institute
Dave Mason, U.S. Army's Communications and Electronics Command (CECOM)

September 2005

Software Architecture Technology Initiative

Technical Note
CMU/SEI-2005-TN-027

This work is sponsored by the U.S. Department of Defense.

The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (http://www.sei.cmu.edu/publications/pubweb.html).

Contents

List of Figures

List of Tables

Acknowledgements

We thank Lawrence Jones and Liam O'Brien of the Carnegie Mellon® Software Engineering Institute for their thorough review of this technical note.

® Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

Abstract

The software architecture of a software-intensive system greatly determines system quality. Evaluating that architecture for fitness of purpose before the system is implemented or undergoes a major modification is a cost-effective approach to uncovering deficiencies early and reducing risk. When used appropriately, software architecture evaluations can have a favorable effect on a delivered or modified government system.

This technical note describes the application of the SEI Architecture Tradeoff Analysis Method® (ATAM®) to the U.S. Army's Warfighter Information Network-Tactical (WIN-T) system. The WIN-T system is being developed by a government-contractor team headquartered at the U.S. Army's Communications and Electronics Command (CECOM) in Ft. Monmouth, New Jersey. This technical note presents the WIN-T program context, the definition of software architecture, and the background of the WIN-T organization and system being evaluated. It also provides a general overview of the ATAM process, describes the application of the ATAM to the WIN-T system, presents important results, and summarizes the benefits the program received.

1 Introduction

Because software architecture is a major determinant of software quality, it follows that software architecture is critical to the quality of any software-intensive system [Clements 96]. For a Department of Defense (DoD) acquisition organization, the ability to evaluate software architectures before they are realized in finished systems can substantially reduce the risk that the delivered systems will not meet their quality goals.

To meet this need for architecture evaluation, the Carnegie Mellon® Software Engineering Institute (SEI) developed the Architectural Tradeoff Analysis Method® (ATAM®) and validated its usefulness in practice [Kazman 00]. This method not only permits evaluation of specific architecture quality attributes (e.g., modifiability, performance, security, and reliability) but also allows engineering tradeoffs to be made among possibly conflicting quality goals.

This technical note describes an ATAM evaluation of the software architecture of the Warfighter Information Network-Tactical (WIN-T) system, which is a sophisticated communications network. This system is being developed by a government-contractor team led by the U.S. Army's Communications and Electronics Command (CECOM) at Fort Monmouth, New Jersey.

Following this introduction, Section 2 defines software architecture, explains the importance of architecture evaluation, and provides an overview of the WIN-T system. Section 3 contains an overview of the ATAM including its purpose and primary steps. Section 4 describes how the ATAM was applied specifically to WIN-T and the results of that application. Section 5 describes some of the post-ATAM activities that resulted from the evaluation, and Section 6 summarizes the overall evaluation.

2 Context for the Architecture Evaluation

2.1 Software Architecture

> *The software architecture of a program or computing system is the structure or structures of the system, which comprise software elements, the externally visible properties of those elements, and the relationships among them* [Bass 03].

The software architecture of a system embodies the earliest software design decisions. These decisions enable or preclude achievement of desired system qualities, such as reliability, modifiability, security, real-time performance, and interoperability. The architecture also forms the basis for the development approach and acts as the blueprint that guides the teams building the system. Architectural decisions are the most critical to get right and the most difficult to change downstream in the development life cycle. The right software architecture paves the way for successful system development. The wrong architecture will result in a system that fails to meet critical requirements, suffers from budget and schedule overruns, and incurs high maintenance costs.

Modern approaches to software architecture take a multi-view approach. A *view* is a representation of one or more structures present in the system. If we consider the analogy of the architecture of a building, various stakeholders (such as the construction engineer, the plumber, and the electrician) all have an interest in how the building is to be constructed. Although they are interested in different components and different relationships, each of their views is valid. Each view represents a structure that maps to one of the construction goals of the building, and these multiple views are necessary to represent the architecture of the building fully. Similarly, a software architecture has a variety of stakeholders, including developers, maintainers, testers, integrators, system administrators, project managers, analysts, certification authorities, end users, and the acquisition organization. Each of these stakeholders has a vested interest in different system properties and goals that are represented by different structural views of the system. The views provide the basis for reasoning about the appropriateness and quality of the architecture for achieving system quality goals.

Some common architectural views include [Clements 02a]

- the logical view, which represents system functions; key system abstractions and their dependencies; and data flows
- the module decomposition view, which represents the hierarchical decomposition of the system's functionality into units of implementation. This decomposition can include objects, procedures, functions, and their relationships.

- the communicating-processes view, which represents processing threads, their synchronization, and the data flows between them
- the deployment view, which shows how the software is allocated to hardware including processors, storage, and external devices or sensors, along with the communications paths that connect them

Other important software architectural views are described in *Documenting Software Architecture: Views and Beyond* [Clements 02a]. Views serve as the primary vehicle for communicating the architecture to its stakeholders, the primary engineering handle for designing quality attributes into the system, and the primary window into the architecture for evaluators who are checking it for fitness of purpose.

Because the architecture is important in system development, it is also important in system acquisition. It is the responsibility of the acquisition organization to ensure that the architecture will support attainment of system quality goals. Formal architecture evaluation is an essential part of an architecture-based acquisition effort, as is insuring that high-quality architecture documentation is produced and maintained.

2.2 The WIN-T Organization

The acquisition organization for WIN-T is CECOM at Fort Monmouth, New Jersey. The Program Executive Office (PEO) Command, Control, and Communications Tactical (C3T), which is collocated at Fort Monmouth, is the management entity responsible for WIN-T. Under the PEO, the day-to-day execution is managed by the program manager of WIN-T (PM, WIN-T), who, in turn, is supported by a project director (PD) and project team. That team is augmented and assisted by various support divisions of the CECOM and PM, WIN-T organization. Figure 1 depicts the WIN-T organizational infrastructure at the time the ATAM-based architecture evaluation was held. Since then, CECOM, PEO C3T, and PEO Intelligence, Electronic Warfare, and Sensors (IEWS) and their subordinate elements have merged into the Communications Electronics Life Cycle Management Center (CE LCMC).

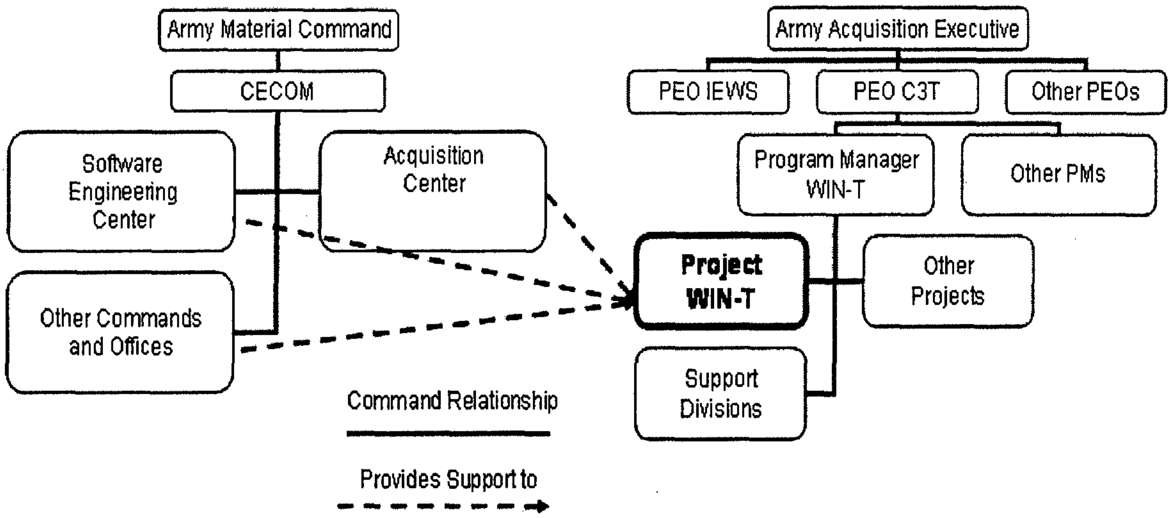


Figure 1: WIN-T Organizational Infrastructure

Three organizations within CECOM provide matrix support to the PM, WIN-T: (1) the Acquisition Center, (2) the Battle Space Systems Division of the Software Engineering Center (SEC), and (3) the Logistics Readiness Command (LRC). Additional external support is provided by the Training and Doctrine Command (TRADOC) Systems Manager (TSM) who is assigned to WIN-T (TSM, WIN-T). The TSM, WIN-T resides at the Fort Gordon Signal Center in Georgia and is the end-user representative responsible for the development of the Operational Requirements Document (ORD).

Within CECOM, the Acquisition Center supports a large number of CECOM organizations including the SEC and a number of PEOs, including PEO C3T. The Acquisition Center, which is home to the contracting officer (KO), provides a full spectrum of acquisition services. Major commodities include aviation communications, man-portable radios, radar systems, computers, satellite communications, night-vision equipment, command-and-control systems, sensors, information management systems, battery and power sources, intelligence/electronic warfare systems, mines/countermines, facilities supplies, and a host of technical services that support the various mission responsibilities of the center's customers.

The SEC provides life-cycle software products and services from the battle space through the sustaining base. The Battle Space Systems Division is the organizational element within the SEC that is responsible for providing centralized software life-cycle engineering, management, and support for more than 150 mission-critical defense systems, including embedded matrix software support to the PM, WIN-T.

2.3 The WIN-T System

WIN-T is the Army's tactical telecommunications system consisting of communication infrastructure and network components designed to provide secure communications at all echelons from the Maneuver Battalion to the Theater Rear boundary. WIN-T is required to be the high-speed, high-capacity, high-mobility backbone communications network, providing voice, data, and video service, for what is referred to as the Army's "Future Force." WIN-T is to set standards and protocols for the Future Force while interfacing with and/or replacing equipment in the Army's current forces and interim Stryker forces. These forces include the Mobil Subscriber Equipment (MSE) that provides communications from Corps to Brigade and the Tri-Service Tactical Communications System (TRI-TAC) that provides communications at Echelons Above Corps (EAC). Moreover, WIN-T will provide command centers and staff elements at the Unit of Employment (UE) with the communications capabilities to link to adjacent UEs, subordinate Maneuver Brigades/Units of Action (MBs/UAs), and sustaining base, Joint, Allied, and Coalition forces. WIN-T is to provide required *reach,*[1] *reachback,*[2] and network operations for the MBs/UAs and interface seamlessly with the Joint Tactical Radio System (JTRS) that extends to the individual warfighter platform.

The operational concept graphic shown in Figure 2 on page 17 depicts how the WIN-T network interconnects UE elements (at Division, Corps, and Theater echelons), as well as the WIN-T presence in UAs within the Future Combat System (FCS)—the network servicing echelons at levels below those served by WIN-T. The figure also shows that WIN-T, as an extension of the Global Information Grid (GIG), provides "reachback" to the continental United States (CONUS). Other external interfaces to current, Joint, Allied, and Coalition forces—the networks adjacent to WIN-T—are also depicted in Figure 2.

WIN-T will transition the Army from a semi-mobile, switched, digital channel system to a fully mobile network that is based on the Internet Protocol (IP) and provides integrated voice, data, and video. It will provide automated, ad hoc management "on the move" with management and services in support of, and controlled by, the commander's policy.

2.4 Contractual Aspects of the Software Architecture Evaluation

WIN-T has been designated an Acquisition Category (ACAT) 1D program, which is the acquisition category for programs in excess of $2.19 billion in procurement funding. WIN-T will actually be well in excess of that amount and, as of this writing, is in the System Design and Development Phase between Milestones B and C. These milestones are decision points on the acquisition path with Milestone B being the decision to proceed with systems design and development and Milestone C being the decision to begin production. The WIN-T

---

1 *Reach* is defined as the ability to communicate with similar or related units, within the theater of operations and beyond the range of organic communications capabilities.

2 *Reachback* is defined as the ability to communicate back to the sustaining base or home station, outside the theater of operations.

development effort was awarded initially as two separate, competing contracts to General Dynamics and Lockheed Martin with the intent to down select to one contractor at about the time of Milestone C. Two years after contract award (and approximately one year after Milestone B), WIN-T underwent a contractual transition that resulted in the elimination of the "competitive fly-off" arrangement. For a variety of reasons, the individual contractual efforts were merged under a single contract with General Dynamics as the prime contractor and Lockheed Martin as a major subcontractor responsible for approximately 50% of the effort. Within this structure, known as the Combined Team, Lockheed Martin has the lead for software development and integration. Each contractor has a software architect, and the General Dynamics architect is designated as the lead software architect. This new single-contract arrangement resulted in a larger number of development contractors participating in the architecture evaluation. It also enabled the stakeholder interests of both major contactors to be accommodated and promoted closer collaboration within the Combined Team.

It was in this context that the Army's Strategic Software Improvement Program (ASSIP) decided to provide funding for a software architecture evaluation of WIN-T, which was to be led by the SEI and based on the ATAM (described in the next section).

Because a software architecture evaluation was not part of the existing WIN-T contract, the PD had to approve the evaluation before proceeding. The government's chief software architect met the PD, WIN-T and explained the process and benefits of an ATAM evaluation. The PD subsequently approved the ATAM evaluation, provided that the contractors were willing to support this effort and that the contractual costs associated with conducting the software architecture evaluation were not excessive.

The next step was to obtain the cooperation of the contractors. Information was provided to the managers and leaders of both contractor organizations via phone, email, and links to the ATAM section of the SEI Web site regarding the process, benefits, and output of an ATAM evaluation. Both contractor organizations were equally enthusiastic about performing an ATAM evaluation.

Before final approval could be obtained, the cost and schedule impact of diverting effort from the planned schedule of events to allow for the ATAM evaluation to take place had to be addressed. With regard to the impact, all the affected parties felt that the potential return of conducting a software architecture evaluation using the ATAM more than justified altering the planned schedule of events and the additional work required on the part of the participating contractors and WIN-T stakeholders. The other costs were for labor, travel, and lodging to have the contractor stakeholders prepare for and participate in the ATAM evaluation. The cost for travel and lodging was estimated at $2,000. The contractual effort to prepare and participate in the ATAM evaluation was estimated to be 200 hours. This included time to prepare and present the required briefings and the participation time spent by the lead architect, four software engineers, and two system engineers representing both contractor organizations.

The KO, WIN-T modified the existing Task Execution Plan (TEP) to support the evaluation. The changes to the TEP did not include the cost of developing and documenting the software architecture, as those tasks were already contract requirements. In fact, the first draft of the Combined Team software architecture had just been delivered. The PD felt that the schedule and cost impact were entirely reasonable and affordable for an ACAT-1D program and gave final approval to proceed with the evaluation.

Once approval to proceed was granted, arrangements were made to have stakeholders from internal offices within the PM (e.g., eventual software maintainers from the CECOM SEC) and stakeholders from external agencies (e.g., TRADOC TSM) participate in the ATAM to ensure that the interests of the WIN-T end users were adequately represented. No difficulties were encountered, and all stakeholders were enthusiastic about the opportunity to have their concerns and architectural issues aired.

3 The ATAM

The purpose of the ATAM is to assess the consequences of architectural decision alternatives in light of quality attribute requirements [Kazman 00]. The major goals of the ATAM are to

- elicit and refine a precise statement of the architecture's driving quality attribute requirements
- elicit and refine a precise statement of the architectural design decisions
- evaluate the architectural design decisions to determine if they address the quality attribute requirements satisfactorily

The ATAM is predicated on the fact that an architecture is suitable (or not suitable) only in the context of specific quality attributes that it must impart to the system. The ATAM uses stakeholder perspectives to produce a collection of scenarios that define the qualities of interest for the particular system under consideration. Scenarios give specific instances of usage, performance requirements, growth requirements, various types of failures, various possible threats, and various likely modifications. Once the important quality attributes are identified in detail, the architectural decisions relevant to each one can be illuminated and analyzed with respect to their appropriateness.

The steps of the ATAM are carried out in two main phases. In the first phase, the evaluation team interacts with the system's primary decision makers: the architect(s), manager(s), and perhaps a marketing or customer representative. During the second phase, a larger group of stakeholders is assembled, including developers, testers, maintainers, administrators, and users. The two-phase approach insures that the analysis is based on a broad and appropriate range of perspectives.[3]

Phase 1:

1. **Present the ATAM.** The evaluators explain the method so that those who will be involved in the evaluation have an understanding of the ATAM process.
2. **Present business drivers.** The appropriate system representatives present an overview of the system, its requirements, business goals, context, and the architectural quality drivers.
3. **Present architecture.** The system or software architect (or another lead technical person) presents the architecture.

[3] These two phases are sandwiched by two less intensive phases. Phase 0 is a preparation phase in which the evaluation activities are planned and set up. Phase 3 is a follow-up phase in which the final report is produced and opportunities for improving the process are considered.

4. **Catalog architectural approaches.** The system or software architect presents general architectural approaches to achieve specific qualities. The evaluation team captures a list and adds to it any approaches they saw during Step 3 or learned during their pre-exercise review of the architecture documentation. For example, "a cyclic executive is used to ensure real-time performance." Known architectural approaches have known quality attribute properties that will help in carrying out the analysis steps.
5. **Generate a quality attribute utility tree.** Participants build a utility tree, which is a prioritized set of detailed statements about what quality attributes are most important for the architecture to achieve (such as performance, modifiability, reliability, or security) and specific scenarios that express these attributes.
6. **Analyze architectural approaches.** The evaluators and the architect(s) map the utility tree scenarios to the architecture to see how it responds to each one.

Phase 2:

Phase 2 begins with an encore of the Step 1 ATAM presentation and a recap of the results of Steps 2 through 6 for the larger group of stakeholders. Then these steps are followed:

7. **Brainstorm and prioritize scenarios.** The stakeholders brainstorm additional scenarios that express specific quality concerns. After brainstorming, the group chooses the most important ones using a facilitated voting process.
8. **Analyze architectural approaches.** As in Step 6, the evaluators and the architect(s) map the high-priority brainstormed scenarios to the architecture.
9. **Present results.** A presentation is produced that captures the results of the process and summarizes the key findings that are indicative of what will be in the final report (a product of Phase 3).

Scenario analysis produces the following results:

- a collection of sensitivity and tradeoff points. A *sensitivity point* is an architectural decision that affects the achievement of a particular quality. A *tradeoff point* is an architectural decision that affects more than one quality attribute (possibly in opposite ways).
- a collection of risks and non-risks. A *risk* is an architectural decision that is problematic in light of the quality attributes that it affects. A *non-risk* is an architectural decision that is appropriate in the context of the quality attributes that it affects.
- a list of current issues or decisions not yet made. Often during an evaluation, issues not directly related to the architecture arise. They may have to do with an organization's processes, personnel, or other special circumstances. The ATAM process records these issues, so they can be addressed by other means. The list of decisions not yet made arises from the stage of the system life cycle during which the evaluation takes place. An architecture represents a collection of decisions. Not all relevant decisions may have been made at the time of the evaluation, even when designing the architecture. Some of these

decisions are known to the development team as having not been made and are on a list for further consideration. Others are news to the development team and stakeholders.

Results of the overall exercise also include the summary of the business drivers, the architecture, the utility tree, and the analysis of each chosen scenario. All of these results are recorded visibly so all stakeholders can verify that they have been identified correctly.

The number of scenarios analyzed during the evaluation is controlled by the amount of time allowed for the evaluation, but the process insures that the most important ones are addressed.

After the evaluation, the evaluators write a report documenting the evaluation and recording the information discovered. This report also documents the framework for ongoing analysis discovered by the evaluators. Clements, Kazman, and Klein provide detailed descriptions of the ATAM process [Kazman 00, Clements 02b].

4 The ATAM Evaluation of WIN-T

4.1 Background

The liaison between the ATAM evaluation team leader and the WIN-T project was the government's lead software engineer for WIN-T. Together, they coordinated the dates for the Phase 1 and Phase 2 meetings, agreed on which stakeholders to invite to each, worked out the delivery of documentation to the team for pre-evaluation review, and worked to make sure that the Step 2 (business drivers) and Step 3 (architecture) presentations were prepared and contained the appropriate information.

Phase 1 took place on February 1-2, 2005 and was followed by Phase 2 on February 8-9, 2005. The evaluation team consisted of four members from the SEI, plus two Army members who had qualified for ATAM participation by successfully completing ATAM training and receiving the SEI ATAM Evaluator Certificate. The evaluation team members' organizations and roles are shown in Table 1.[4]

Table 1: Evaluation Team Members

Organization	Role
SEI	Team leader, evaluation leader
SEI	Questioner, scribe
SEI	Timekeeper, questioner
SEI	Data gatherer, questioner
Army, PEO C3T	Questioner, process observer
Army, Research, Development, and Engineering Command (RDECOM) CERDEC Software Engineering Directorate (SED)	Questioner, process enforcer

The system stakeholders (architects, managers, developers, testers, integrators, etc.) participating in the WIN-T ATAM evaluation exercise are identified in Table 2 and Table 3.

4 All participants' names have been withheld for privacy reasons.

Table 2: Attendees for Phase 1 of the WIN-T ATAM Evaluation

Organization	Role
General Dynamics	Systems Engineering NetOps Integrated Product Team (IPT)
PM, WIN-T (SEC)	Lead Government Software (SW) Engineer
PM, WIN-T	Project Director
PM, WIN-T	Information Assurance (IA) Team Leader
General Dynamics	Team Lead Software Architect
PM, WIN-T (SEC)	Software Engineer
Lockheed Martin	Team Lead Software Engineer
Lockheed Martin	Software Developer
General Dynamics	General Dynamics Lead Software Engineer
Lockheed Martin	Software Developer

Table 3: Attendees for Phase 2 of the WIN-T ATAM Evaluation

Organization	Role
Lockheed Martin	Systems Architect
General Dynamics	Systems Engineering NetOps IPT
CERDEC SED	Software Supportability
PM, WIN-T (SEC)	Lead Government SW Engineer
CECOM SEC (L3 ILEX)	User Representative
PM, WIN-T	IA Team Leader
PM, WIN-T	Program Analyst
CECOM SEC	Software Supportability
PM, WIN-T (Tecolote Research)	Cost Estimating
General Dynamics	Team Lead Software Architect
PM, WIN-T (SEC)	Software Engineer
TSM, WIN-T	User Representative

Table 3: Attendees for Phase 2 of the WIN-T ATAM Evaluation (cont'd.)

Organization	**Role**
PM, WIN-T	Logistics
Army Research Lab	PM, WIN-T Human Factors Engineering (HFE) support
PM, WIN-T	Logistics Lead
PM, WIN-T	Network Engineer
General Dynamics	Lead Software Engineer
PM, WIN-T (Space and Terrestrial Communications Directorate [S&TCD])	Deputy PD
PM, WIN-T	Risk Manager
Lockheed Martin	Team Lead Software Engineer
Lockheed Martin	Software Developer

4.2 Business and Mission Drivers

Step 2 of the ATAM method is a presentation of the system's business and mission drivers. Before the exercise, the evaluation leader works with the person making the presentation and provides him or her with a standard presentation outline and template, to make sure the desired information is produced. The goal of the one-hour presentation is to understand why (from the development side as well as the acquisition side) this system is being created. For government acquisition programs, the person making the presentation is usually from the program office for the system being acquired. The purpose is to start collecting quality attribute goals against which the architecture can be evaluated.

For the WIN-T evaluation, the Army's PD, WIN-T gave an overview of WIN-T and described the Army's business objectives for the program. The driving business and mission requirements for the WIN-T products and the architecture goals derived from these requirements included the points described below.

4.2.1 General Points

1. The purpose of the WIN-T system is to provide a single integrating communications network to (1) function as the Army's tactical portion of the GIG and (2) to link FCS with higher Army echelons and the GIG
2. WIN-T is to provide a reliable and secure network with high speed, high capacity, and high quality of service (QoS).

3. The WIN-T system is required to support mobile communications and provide a replacement communications architecture for legacy systems such as MSE, TRI-TAC, Integrated Systems Control (ISYSCON), and Trojan Spirit.
4. The Win-T deployment level is Theater to Maneuver Battalion. It will be owned, operated, and maintained by both signal and non-signal units.
5. The development schedule is aggressive with the Initial Operational Test (IOT) scheduled for September 8, 2005. Spiral development will be used to achieve the initial capability.

4.2.2 Points Related to Business Goals

1. Deploy a user-friendly human machine interface (HMI) for the WIN-T operator.
2. Minimize the amount of network management the WIN-T operator must perform (automate the planning process to the maximum extent possible).
3. WIN-T must be operable and maintainable within the budget and schedule.
4. WIN-T must be transparent to the combat users.
5. The operation of the software must be transparent to the WIN-T operators.
6. The maintenance of the software must be as transparent as possible to the WIN-T operators.

4.2.3 Points Related to Key Performance Parameters

Key performance parameters for WIN-T are listed in Table 4.

Table 4: Key Performance Parameters (KPPs) for WIN-T

Key Performance Parameter	Threshold	Objective
Interoperability	100% of critical information exchange requirements	Meet all information exchange requirements.
Network Reliability (Probability)	.98 (at the halt) .90 (mobile)	.99 (at the halt) .97 (mobile)
Network Management	Manage components from physical location, in area of responsibility	Manage components from virtual location, outside area of responsibility.
Information Dissemination	<= 5 seconds (critical survival information) < 8 seconds (time-sensitive information)	< .5 seconds (critical survival information) < 1 seconds (time-sensitive information)
Information Assurance	Protect/defend against 95% [Block 1], 98% [Block 2] and 99% [Objective] of all attacks from known/external threats.	Protect/defend against 99% of all attacks from known/external threats
Mobile Throughput	Traveling at 25 mph with 256 Kbps throughput (ground speed)	Traveling at 45 mph with 4 Mbps throughput (ground speed)

4.2.4 Points Related to Business Constraints

Business constraints affecting WIN-T include

- long life cycle
  - fielding from 2008 through 2020
  - expected lifetime of 20+ years
- interoperable
  - Army Enterprise network (echelon above WIN-T)
  - FCS (echelon below WIN-T)
  - Joint networks
  - Coalition network
- upgradeable
  - accommodation of possible early spirals (less capable early versions)
  - planned block upgrades (added capabilities)
  - market-driven, commercial off-the-shelf (COTS) upgrades

- security upgrades

4.3 Architecture Presentation

Before the evaluation, the evaluation leader works with the architect to prepare an architecture presentation for the evaluation exercise. The presentation lasts between one and two hours and focuses on (1) a set of architectural views and (2) the primary approaches used to satisfy the architecturally significant requirements and driving quality attributes.

The views presented included

- a context view
- a layered view
- a deployment view
- a functional or module decomposition view
- one or more component-and-connector views

4.3.1 Context View

The context view (Figure 2) puts WIN-T into perspective with respect to its users and other participating elements.

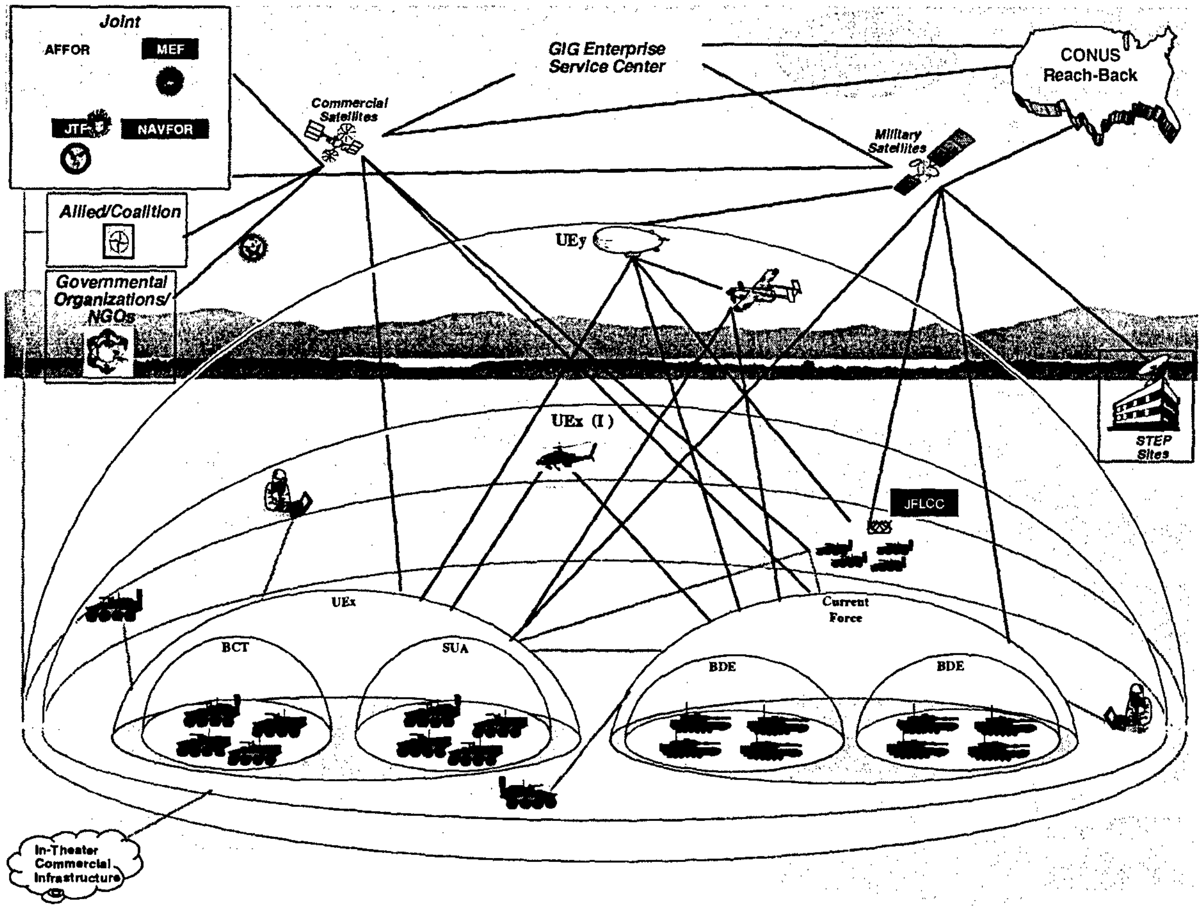


Figure 2: Context Diagram (OV-1)

4.3.2 Layered View

The layered view is shown in Figure 3. The layer concept is used to help bring properties of modifiability and portability to a software system. A layer is an application of the principle of information hiding.

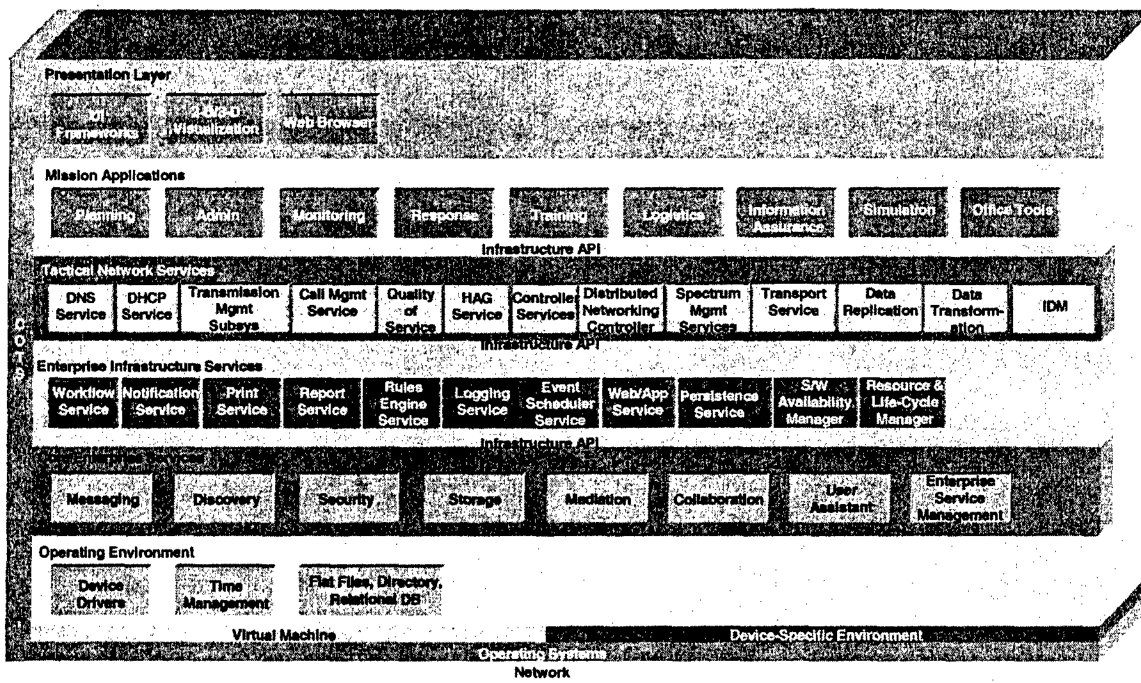


Figure 3: Layered View

4.3.3 Deployment View

Figure 4 shows a deployment view that illustrates the allocation of software to hardware.

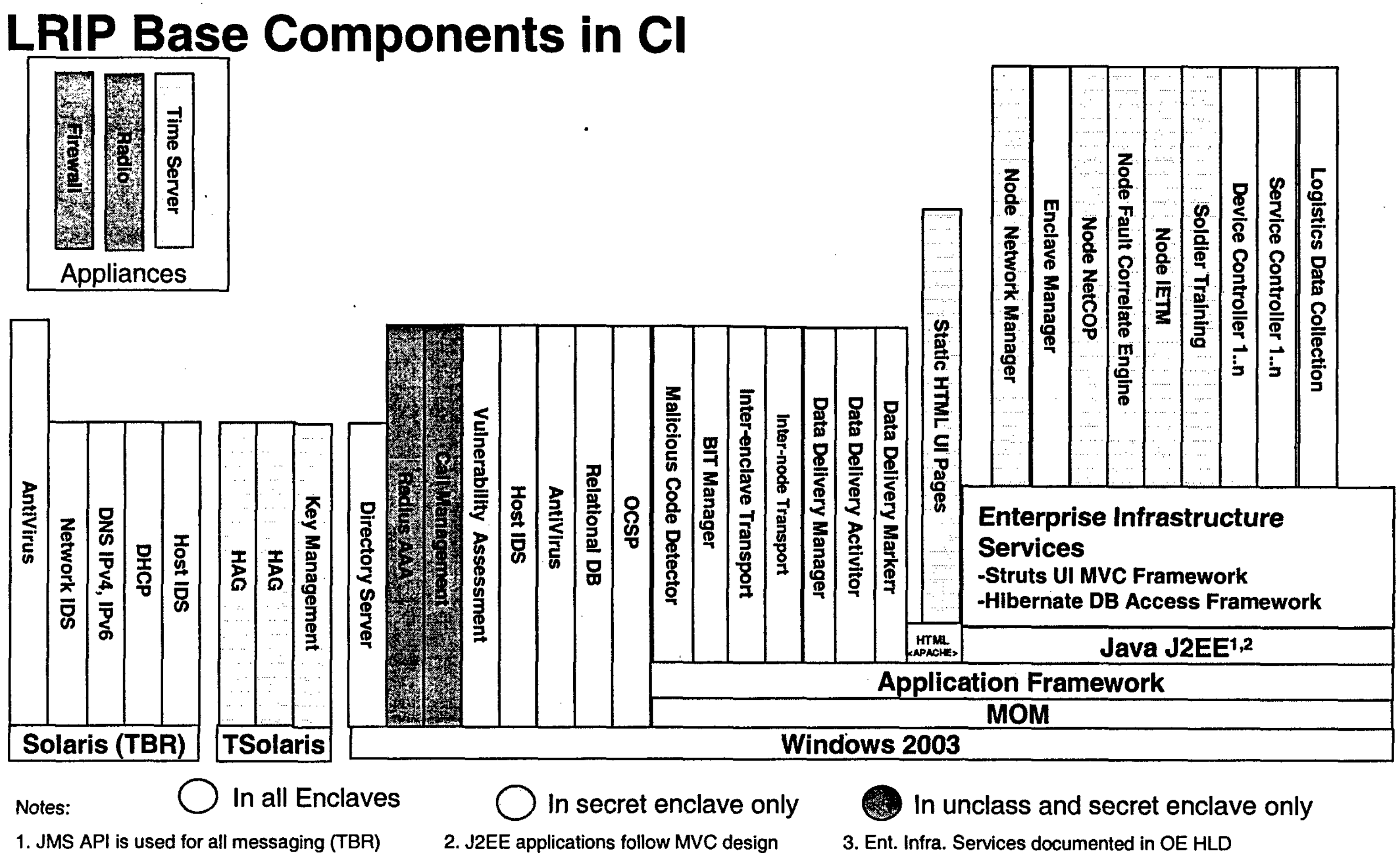


Figure 4: Deployment View

4.3.4 Functional or Module Decomposition View

The functional view for WIN-T has six computer software configuration items (CSCIs) with the relationships among them shown in Figure 5. The NetOps (Network Operations) functional area includes NetOps Management (NM), Information Assurance (IA) management, Information Dissemination Management (IDM), and Network Services (NS), all running in the Operational Environment (OE). The Transmission Subsystem (TS) is software and firmware that runs on the WIN-T radios.

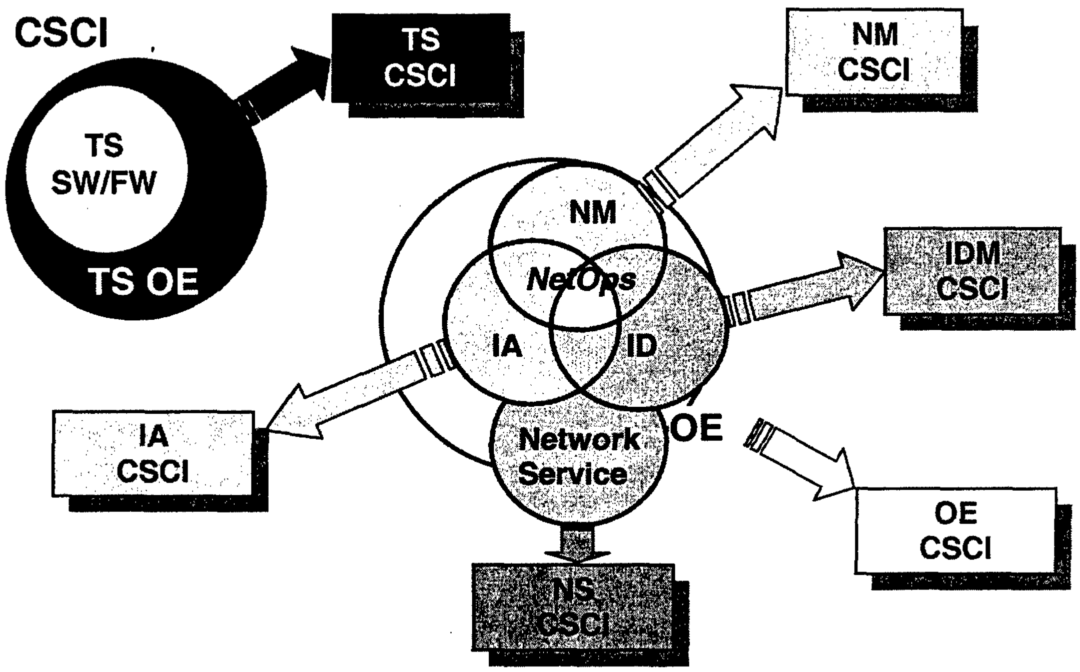


Figure 5: Functional View

Module decomposition views can consist of several levels of decomposition; the finest grained levels indicate the smallest units of implementation that might be affected by a modification to the system. Figure 6 is an example hierarchical functional decomposition for one CSCI. In this figure, Training is shown as a descendant of the CSCI, whereas it is actually a descendant of the unit labeled "Operational Environment."

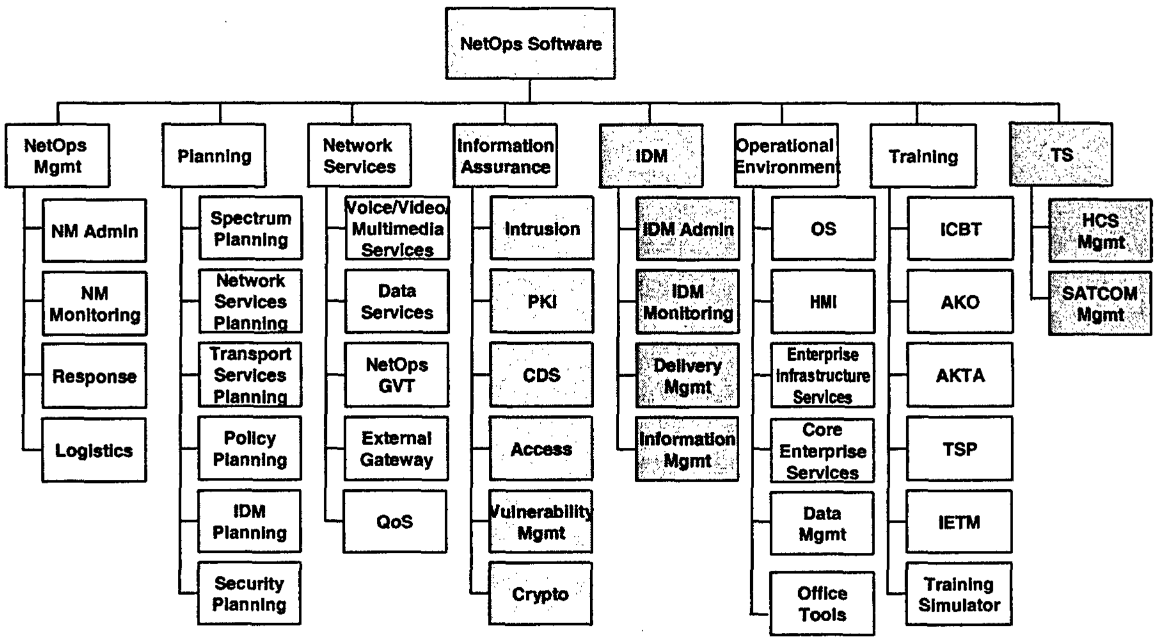


Figure 6: Decomposition View

4.3.5 Component-and-Connector View

One or more component-and-connector views show the runtime interaction of computational elements and communication pathways. Figure 7 shows how the major functional elements communicate via a message bus.

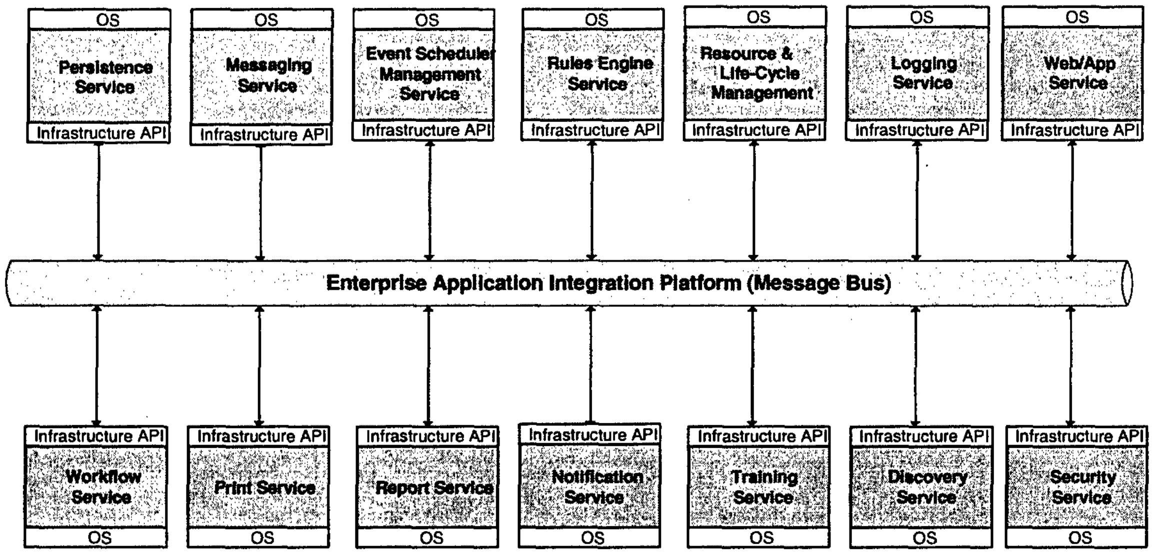


Figure 7: Component-and-Connector View

Figure 8 shows the replication/failover scheme for fault tolerance and high availability.

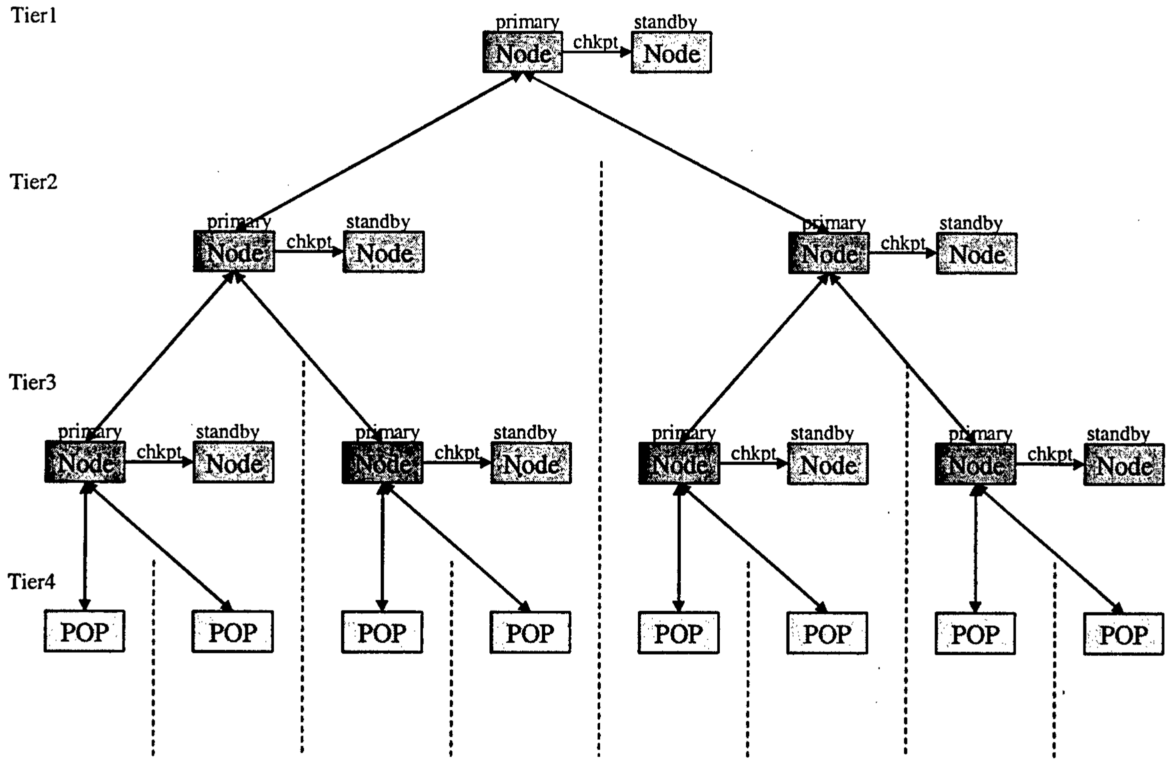


Figure 8: Replication/Failover Scheme

4.4 Architectural Approaches

After the architecture is presented, the evaluation team summarizes the architectural approaches that have been employed. The team compiles the list by reviewing the architecture documentation ahead of time and extracting the approaches mentioned in the architecture presentation.

Although the WIN-T architecture employs many approaches, the five main approaches described in Table 5 were identified.

Table 5: Architectural Approaches

Approach	Description	Architecture Layer
1. Enterprise Application Integration Platform	The WIN-T Enterprise Application Integration Platform (Message Bus) is a combination of a common data model, a common command set and application program interface (API) (Java Message Service [JMS]) and a messaging infrastructure to allow different systems to communicate through a shared set of interfaces.	Core Enterprise Services Layer
2. Service-Oriented Architecture (SOA)	The WIN-T SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents.	All Service Layers
3. Infrastructure API	An Infrastructure API is a large collection of ready-made software components that provide many useful capabilities, such as service-oriented classes and methods. The WIN-T API is language agnostic and grouped into libraries of related classes and interfaces; these libraries are known as packages.	All Service Layers
4. Enterprise Information Dashboard (EID) and User Interface (UI) Frameworks.	The WIN-T EID Architecture represents a Web site that provides a single point of access (Single Sign-On) to applications and information and may be one of many hosted within a single WIN-T server. UI Frameworks like the Model View Controller (MVC) pattern was designed to decouple the graphical interface of an application from the code that actually does the work.	HMI Layer
5. Workflow Engine	One of the key WIN-T services is the workflow engine that provides the capability to orchestrate and collaborate homogeneous and disparate services using the Business Process Execution Language (BPEL) standard.	Enterprise Infrastructure Services

These architectural approaches are illustrated in Figure 9.

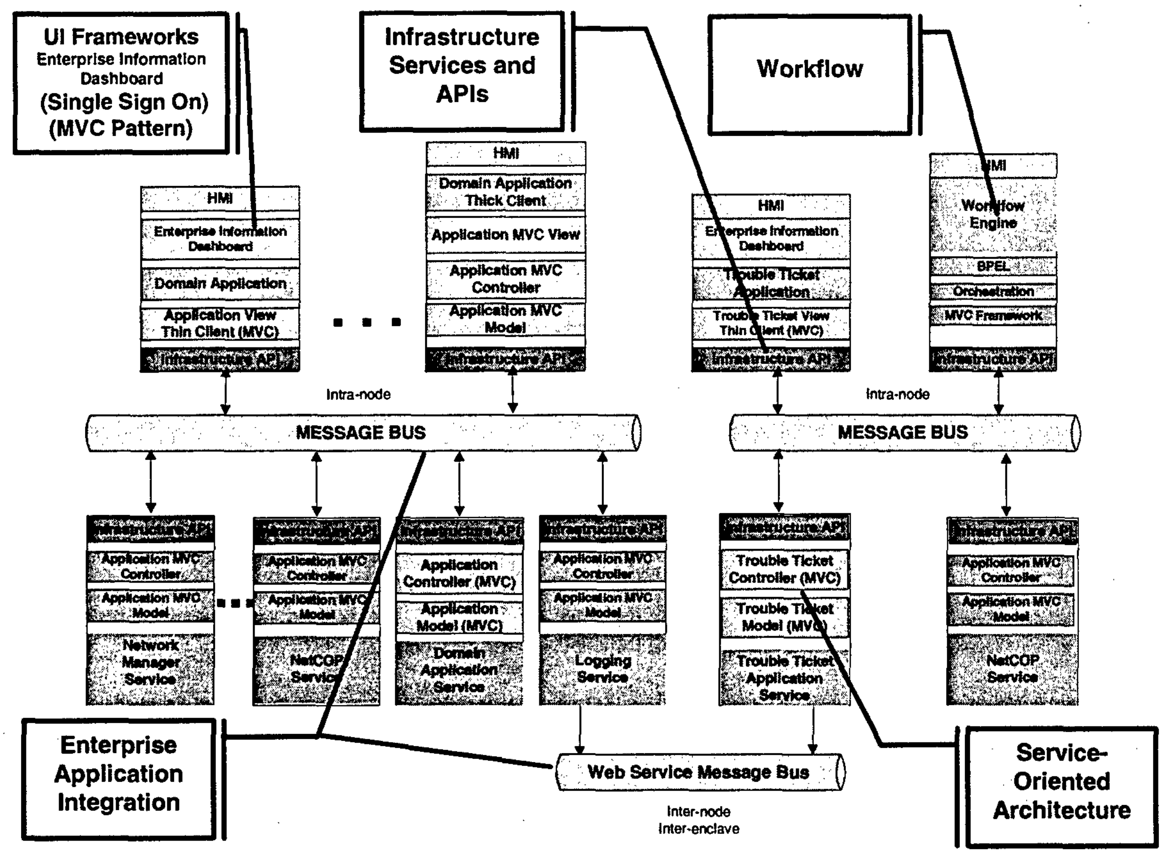


Figure 9: Architectural Approaches

4.5 Utility Tree

The utility tree provides a vehicle for translating the quality attribute goals articulated in the business drivers presentation to scenarios that express quality attribute requirements in a form specific enough for them to be "tested" against the architecture. The stakeholders who are present at Phase 1 construct the utility tree under the facilitated guidance of the ATAM team leader.

In this tree, "Utility" is the root node and expresses the overall "goodness" of the system. In the case of WIN-T, the second level nodes were modularity, adaptability, resilience, usability, security, performance, interoperability, information dissemination, autonomous operation, standards compliance, scalability, flexibility, maintainability, supportability, survivability, mobility, and affordability.

Under each of these quality attributes are specific concerns. These concerns arise from considering the quality-attribute-specific stimuli and responses that the architecture must address. For example, in the case of WIN-T, modularity was decomposed into the following concerns:

- replacement of an architectural component

- loose coupling/high cohesion
- separately implementable
- well-defined interfaces
- designed from a common set of components

Finally, these concerns are characterized by a small number of scenarios. These scenarios become leaves of the utility tree; thus the tree has four levels.

A scenario represents a use, or modification, of the architecture, applied not only to determine if the architecture meets a functional requirement but also (and more significantly) to predict system qualities such as performance, reliability, modifiability, and so forth.

The scenarios at the leaves of the utility tree are prioritized along two dimensions:

1. importance to the system
2. perceived risk in achieving the particular goal

These nodes are prioritized relative to each other using ranking pairs of High, Medium, and Low (H, M, L), where the first value in the pair indicates the degree of importance to the system and the second indicates the degree of difficulty for the architecture to achieve it.

The quality attribute utility tree elicited during the WIN-T evaluation is reproduced in Table 6. It is typical in size to utility trees collected at other ATAM-based evaluations. Not all the attribute concerns are filled in with elaborating scenarios. This is normal and reflects the fact that sometimes stakeholders can think of a broad description of a quality attribute but not a specific requirement for it. The scenarios listed in Table 6 are numbered in the order in which they were created.

Table 6: WIN-T Quality Attribute Utility Test[5]

Quality Attribute	I. Modularity	
Attribute Concerns	A. Replace an architectural component	
Scenarios	*1. Replace DBMS with a new version during maintenance and accomplish replacement within 1 person month.*	(M,L)
Attribute Concerns	B. Loose coupling/high cohesion	
Attribute Concerns	C. Separately implementable	
Scenarios	*3. A contractor at one site implements a service that uses a service developed at a different site during development. Contractor implements the service knowing only the interface definition of the used service.*	(H,H)

[5] The information in this table has not been edited and represents exactly what was captured during the ATAM evaluation. For definitions of acronyms in this table, see Appendix A on page 49.

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Attribute Concerns	D. Well-defined interfaces	
Quality Attribute	II. Modularity (cont'd.)	
Scenarios	*2. NCES-provided version of a WIN-T service becomes available (UDDI) during block upgrade. WIN-T-provided service is replaced by NCES-provided service, any service that uses UDDI does not have to change, and the work takes 1 calendar month.*	(H,M)
Attribute Concerns	E. Design from a common set of components	
Quality Attribute	II. Adaptability	
Attribute Concerns	A. Ability to accommodate new requirements	
Scenarios	*4. There is a requirement to model a new radio during maintenance. Changes to the architecture to accommodate the new radio are localized (ideal solution requires only a database change).*	(H,M)
	7. Add fault correlation capability to WIN-T during maintenance, and the task completed in no more than 6 person months.	(H,H)
Attribute Concerns	B. Ability to accommodate new technologies	
Scenarios	*6. Introduce a wearable head mounted display during maintenance, and modifications are limited to UI modules.*	(L,M)
Attribute Concerns	C. Ability to field a subset of the current requirements (functionality)	
Scenarios	*8. Add block 0.5 NetOps capability during development with minimal infrastructure in 07 time frame*	(H,H)
Attribute Concerns	D. Ability to support various platforms	
Scenarios	*5. There is a requirement to port to a new computer platform with same family of OS during maintenance. No software above the VM is changed; port takes no more than 3 months.*	(M,L)
Quality Attribute	III. Resilience	
Attribute Concerns	A. Ability to accommodate unanticipated user actions	
Scenarios	*9. Power user modifies planning workflow during operations. The system performs sanity check on user's modifications and warns of potential problems; rolls back to known good workflow definition.*	(H,M)
	10. Naïve user attempts to disable firewalls during operations and system refuses to allow changes.	(H,L)

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	IV. Usability	
Attribute Concerns	A. Minimize user training (operator, user, non-signal user)	
Scenarios	*11. Non-signal operator training during unit training takes no more than 4 hours; operator able to perform assigned tasks; requires no classroom training*	(H,M)
	12. A signal user transfers from the 1 CAV to the 4 ID during operations, and the person can use the system immediately with only minimal briefing, embedded training, and on-line help.	(H,L)
	13. Small unit training must be conducted during operations and not impact on operations; minimal reconfiguration.	(H,H)
Attribute Concerns	B. Enable user to perform tasks in allocated time.	
Scenarios	*14. User performs a hasty replan (new frequency allocation) during operations. The system allows hasty plan using parts of existing plans within 15 minutes.*	(H,M)
Attribute Concerns	C. Minimize number of operators.	
Attribute Concerns	D. Consistent GUI (common look and feel)	
Attribute Concerns	E. Minimize operation impact.	
Attribute Concerns	F. Availability of online help, training, and simulation	
Attribute Concerns	G. Support on-the-move, protective gear, at-the-halt operation.	
Scenarios	*15. Switch from at-the-halt to on-the-move operation during operations, and the system allows user to immediately switch GUI.*	(H,L)
Attribute Concerns	H. Communicate execution status	
Quality Attribute	**V. Security**	
Attribute Concerns	A. Authenticate users.	
Scenarios	*16. User wants to log in using an approved authentication means during operations, and equipment recognizes the user and accords him appropriate permissions.*	(H,M)
Attribute Concerns	B. Intrusion detection	
Scenarios	*17. User attempts to access an area he is not authorized to access during operations. User is denied access, and a security alert and audit trail is generated.*	(H,M)

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	V. Security (cont'd.)	
Scenarios (cont'd.)	*18. A unit is overrun during operations. The system detects the unauthorized users and generates a security alert; an authorized person can "zeroize" the system and shut it down remotely.*	(H,H)
Attribute Concerns	C. Virus detection	
Scenarios	*19. Antivirus software initiates a scan during operations, and system continues to operate without degradation while scan is in progress.*	(H,L)
Attribute Concerns	D. User authorization (user = individual or software application)	
Attribute Concerns	E. Nonrepudiation	
Attribute Concerns	F. Information protection	
Attribute Concerns	G. Multiple levels of security	
Scenarios	*20. Operator has to perform management tasks in security domains except TS during operations. System allows operator to manage domains from a single domain without security compromise.*	(H,H)
Attribute Concerns	H. Certifiability and accreditation	
Attribute Concerns	I. Policy based security management	
Scenarios	*21. Commanding General wants to override current security policy to allow a node to broadcast on a certain frequency during operations. The system reports the violation but provides a mechanism to allow the override to be effected.*	(M,M)
Attribute Concerns	J. Selective "zeroization"	
Quality Attribute	VI. Performance	
Attribute Concerns	A. Minimize NetOps bandwidth	
Scenarios	*22. Following a complete replan during peak operational traffic system does not allow management traffic to intrude on the operational bandwidth.*	(H,H)
Attribute Concerns	B. Timely dissemination and activation of network policy changes	
Attribute Concerns	C. Meet IER message latency requirements.	
Scenarios	*23. There is a change in latency requirements for a certain class of traffic during a change in Ops tempo. The system meets new message latency requirements.*	(M,L)
Attribute Concerns	D. Meet IER message completion requirements.	
	24. New messages are added to the critical message list during operations. The system meets the message completion requirements.	(M,L)
Attribute Concerns	E. Planning cycle time	

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	VI. Performance (cont'd.)	
Attribute Concerns	F. Service response time meets operator expectations	
Scenarios	*25. User at NetOps cell performs a significant "replan" during operations. It is completed within 1 hour.*	(H,H)
Attribute Concerns	G. Cold start/warm start/hot start/shutdown latency	
Scenarios	*26. A node is cold started during operations, and the start is completed within 30 minutes.*	(H,H)
Attribute Concerns	H. Situational awareness currency	
Quality Attribute	VII. Interoperability	
Attribute Concerns	A. Ease of interfacing with using applications	
Scenarios	*28. WIN-T hosts JNMS in NOSC-Y, the army is the combatant commander and JNMS software runs.*	(H,H)
	29. WIN-T is hosting JNMS in NOSC-Y, and during maintenance, JNMS wants to change COTS products. Both systems run; minimal impact on code.	(H,H)
Attribute Concerns	B. Ease of interfacing with other systems	
Scenarios	*27. WIN-T interoperates with JNMS, the army is not the combatant commander. The system supports exchange of messages within acceptable time frame using appropriate formats.*	(H,L)
	31. There is a need to interface with a new NCES compliant system on the GIG (obtain medical records) during operation. There is no change to WIN-T.	(H,L)
	30. A non-IDM aware software application runs during operations. Messages are handled in accordance with IDM policies.	(L,H)
Quality Attribute	VIII. Information Dissemination	
Attribute Concerns	A. Right data to the right destination at the right time	
Scenarios	*32. Application has published some situational awareness data during operations, and the data is available within the required time.*	(H,H)
Attribute Concerns	B. Accessibility of information by application processes	
Attribute Concerns	C. Appropriate persistence	
Attribute Concerns	D. Prioritization	

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	IX. Autonomous Operation	
Attribute Concerns	A. System continues to operate without operator intervention	
Attribute Concerns	B. Subsets of nodes able to operate in isolation	
Scenarios	*33. A set of nodes is isolated from the network during operations and nodes are able to interoperate with each other; nodes retain critical data for 72 hours.*	(H,M)
Quality Attribute	X. Standards Compliance	
Attribute Concerns	A. DoD	
Scenarios	*34. WIN-T transitions from compliance with COE and NCES to compliance with NCES only during development with no impact to delivery and cost objectives.*	(H,M)
Attribute Concerns	B. Non-DoD	
Quality Attribute	XI. Scalability	
Attribute Concerns	A. Provide sufficient resource capacity.	
Attribute Concerns	B. Increase number of units supported.	
Scenarios	*35. The number of deployed nodes increases from 500 to 3000+ nodes during operations, and system still meets all performance requirements.*	(H,H)
Attribute Concerns	C. Increase number of users supported	
Scenarios	*48. WIN-T is required to take over management of all edge devices during maintenance. This is accommodated within 1 spiral; maintain performance requirements without increasing footprint.*	(L,H)
Attribute Concerns	D. Increase traffic load	
Attribute Concerns	E. Increase network size	
Attribute Concerns	F. Increase geographic coverage	
Attribute Concerns	G. Increase Ops tempo	
Quality Attribute	XII. Flexibility	
Attribute Concerns	A. Policy based management	
Scenarios	*46. A policy is changed at a higher level node during operations. Lower level nodes automatically disseminate and reconfigure to the new policies.*	(M,H)

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	XII. Flexibility (cont'd.)	
Attribute Concerns	B. Response to changing operational environment	
Scenarios	*36. The Ops tempo changes during operations and proper policies for the new Ops tempo are executed within 5 minutes.*	(H,H)
	37. A node that was "zeroized" has to be rebuilt during operations. The node configuration is restored to the appropriate current state within 30 minutes.	(H,H)
Attribute Concerns	C. Incorporation of unplanned nodes in network	
Quality Attribute	XIII. Maintainability	
Attribute Concerns	A. Ability to accommodate new technologies	
Scenarios	*47. SOAP changes to no longer use XML during maintenance. There is minimal impact on system; accommodated within 1 spiral.*	(L,H)
Attribute Concerns	B. Ability to support isolation of faults	
Attribute Concerns	C. Minimize training for maintainers	
Attribute Concerns	D. Minimize test bed requirements	
Attribute Concerns	E. Ability to download patches	
Attribute Concerns	F. Configuration management and tracking	
Attribute Concerns	G. Backward compatibility	
Quality Attribute	XIV. Supportability	
Attribute Concerns	A. Ability to use organic support for distribution	
Scenarios	*39. A new patch is developed during operations. The patch is distributed over the network without contractor intervention or support.*	(H,M)
Quality Attribute	XV. Survivability	
Attribute Concerns	A. Survive NOSC failure	
Scenarios	*41. NOSC-Y operations cell becomes inoperable during operations, and the planning cell takes over with minimal disruption and within 10 minutes.*	(H,H)
	44. All NOSC-Ys go out during the execution of provisioning, and a NOSC-X takes over and plans and manages the network within 10 minutes; resynchronizes within 30 minutes.	(H,H)

Table 6: WIN-T Quality Attribute Utility Test (cont'd.)

Quality Attribute	XV. Survivability	
Scenarios (cont'd.)	*45. A previously partitioned network reconnects during operations and the two segments synchronize within 30 minutes.*	(H,H)
Attribute Concerns	B. Survive link failure	
Attribute Concerns	C. No single point of failure	
Attribute Concerns	D. Graceful degradation in face of software failure	
Attribute Concerns	E. Fault prediction	
Attribute Concerns	F. Isolate a rogue node	
Attribute Concerns	G. Mitigate denial of service attack	
Attribute Concerns	H. Ability to implement LPI/LPD	
Attribute Concerns	I. Service a (non-NOSC) node failure	
Quality Attribute	XVI. Mobility	
Attribute Concerns	A. Ability to operate on the move	
Attribute Concerns	B. Ability to manage on the move	
Attribute Concerns	C. Ability to plan en route	
Scenarios	*42. A change in the battlefield situation occurs when enroute aboard transport aircraft. The system supports planning, and plan rehearsal and subsequent download to the network.*	(H,M)
Quality Attribute	XVII. Affordability	
Attribute Concerns	A. Ability to manage recurring COTS costs	
Scenarios	*43. A COTS package becomes inordinately expensive and one or more open source options are available during maintenance. Evaluate open source options; pick an option; install replacement for a comparable cost to the original COTS package.*	(H,H)
Attribute Concerns	B. Cost of development environment	

4.6 Scenario Generation and Prioritization

In addition to the scenarios at the leaves of the utility tree, the scenario elicitation process in Step 7 allows the larger group of stakeholders to contribute additional scenarios that reflect their concerns and understanding of how the architecture will accommodate their needs. While the scenarios that appear in the utility tree are developed top down, the scenarios generated by the larger group of stakeholders are developed bottom up. The combination of approaches provides some assurance that high-priority scenarios are surfaced. A particular scenario may, in fact, have implications for many stakeholders: for a modification, one stakeholder may be concerned with the difficulty of a change and its performance impact, while another may be interested in how the change will affect the "integrability" of the architecture.

Table 7 shows the scenarios that were collected by a round-robin brainstorming activity during Step 7 in Phase 2 of the ATAM evaluation. Each scenario is elicited in three parts: (1) a stimulus, describing an interaction of a stakeholder with the system, (2) an environment, describing what activity was ongoing at the time of the stimulation, and (3) a response, indicating the desired outcome, in quantitative terms. Stakeholders were free to choose a scenario from the utility tree as their contribution or create one on their own. After the scenarios were generated, they were prioritized using a voting process in which participants were given 10 votes that they could allocate to any scenario or scenarios they chose. The number of votes each scenario received is shown in the rightmost column of the table. Numbering begins at 49, since there were 48 scenarios previously collected in the utility tree.

Table 7: Phase 2 Scenarios

Scenario #	Stimulus	Environment	Response	Votes
49	A COTS product is becoming increasingly incompatible with other COTS products being used.	development	The COTS product is replaced.[6]	n/a
50	Scenario #43 from utility tree			0
51	WIN-T has the requirement to interface with a lower level system (ISYSCON, TIM).	operational	WIN-T interoperates without any changes to other systems.	10
52	Scenario #34 from utility tree			0
53	Scenario #2 from utility tree			0
54	Scenario #39 from utility tree		Documentation impact	0
55	A non-signal operator/ maintainer needs to identify and replace a faulty piece of equipment.	operational	WIN-T provides an automated capability to assist in troubleshooting at non-signal user level.	5
56	Users need to be able to access the network on the move.	on-the-move operation	WIN-T supports 256 Kbit at 45 mph during move.	5
57	A WIN-T gateway node has been corrupted due to a network intrusion.	operational	Intrusion and corruption is detected and another node takes over duties.	12
58	A legacy application replaces an organic service with a WIN-T service.	legacy system maintenance; WIN-T operational	All documentation and APIs are sufficient; service implementations are available for inclusion in an integration/test environment.	12
59	A software problem is detected that affects all nodes.	operational	Problem is detected, isolated, logged; information is conveyed to maintainers for development of a patch.	12
60	High-priority information needs to be moved across the network, but the current setup prohibits the timely transfer.	operational	WIN-T has to reconfigure itself within the current policies to allow the transfer to occur; reconfigures back at completion	12
61	Signal officer gets an OPORD that requires building a network.	operational	WIN-T provides planning tools to generate the network and the signal annex within required time.	8

[6] During voting, this scenario was combined with scenario #73.

Table 7: Phase 2 Scenarios (cont'd.)

Scenario #	Stimulus	Environment	Response	Votes
62	A new baseline version of the software is distributed to a portion of a force that interoperates.	maintenance and operational	Units with different versions interoperate.	16
63	see 13			5
64	User needs to do enroute training.	enroute to area of operations	Users plan and rehearse scenario in simulation mode.	1
65	A virus gets into a server.	operational	The virus is detected and neutralized.	9
66	WIN-T needs to update virus detection capabilities.	combat operational	System allows users to evaluate impact of downloading new capabilities to operating system; download when appropriate.	9
67	see 18			0
68	A new mapping of IP addresses to force elements has occurred.	operational	The system allows organic personnel to make changes and update the IP addresses.	13
69	A collection of vehicles collects in a "parking lot."	operational	The networks can self-organize and nodes configure themselves within 5 minutes.	19
70	NCES services become available.	maintenance	Developers can evaluate newly available services and switch over where appropriate.	11
71	User changes the device they are using.	operational	System adapts to new device; 1 code base for devices	4
72	Software upgrades impact training, documentation, simulation, and maintenance documentation.	operational	Impacted artifacts updated concurrent with patch	7

Table 7: Phase 2 Scenarios (cont'd.)

Scenario #	Stimulus	Environment	Response	Votes
73	Numerous disparate COTS products have to be integrated.	development	Products are integrated without changing them or the architecture (combined with 49).	18
74	WIN-T needs to support an unanticipated mode or state (unmanned node).	maintenance	New mode or state can be added in without code changes.	7
75	There is some sort of complex network problem that cannot be solved at a low level.	operational	Somebody at a higher level is able to take control of the network and identify and resolve the problem.	8
76	Need to do incremental testing	development	Architecture supports incremental testing of parts of the system.	2
77	QoS and SoS request comes from FCS.	operational	System propagates that request up to the GIG and honors the request as appropriate.	10
78	Add an unmanned node.	operational	Add node and secure it.	0
79	A hardware upgrade occurs.	maintenance	The software accommodates the upgrade and anticipates the need for a hardware upgrade.	6
80	Vehicle starts to move with antenna mast up.	operational	System identifies safety condition; alerts operator.	3

4.7 Overview of the Analysis Process

During analysis (Steps 6 and 8), the ATAM evaluation team facilitates the analysis of the high-priority scenarios. Scenarios are analyzed in detail by walking through each one to evaluate its effects on the architecture. The ATAM evaluation team facilitates the ensuing stakeholder discussion to surface architecturally based risks, sensitivities, and tradeoffs. The stakeholders contribute to the analysis by discussing issues regarding the architecture from their points of view.

Fifteen scenarios from the Phase 1 utility tree and the Phase 2 scenario generation process were examined in detail vis-à-vis the WIN-T architecture. The scenarios that were examined are listed below. Half are scenarios from the utility tree that received an (H,H) priority rating; the other half (intermingled) are the leading vote-getters from Step 7.

- Scenario 69: A collection of vehicles collects in a "parking lot" during operations. The networks can self-organize and nodes configure themselves within 5 minutes.

- Scenario 3: A contractor at one site implements a service that uses a service developed at a different site during development. Contractor implements the service knowing only the interface definition of the used service.
- Scenario 49: (A COTS product is becoming increasingly incompatible with other COTS products being used in development environment. The COTS product is replaced.) Combined with Scenario 73. (Numerous disparate COTS products have to be integrated in the development environment. The products are integrated without changing them or the architecture.)
- Scenario 7: One or more problems are occurring on the network. We need to add fault correlation capability to WIN-T during maintenance to have the capability to identify and correct them.
- Scenario 62: A new baseline version of the software is distributed to a portion of an operational force that interoperates as part of maintenance. Units with different versions interoperate.
- Scenario 13/63:[7] Small unit training must be conducted during operations and not impact on operations; minimal reconfiguration.
- Scenario 68: A new mapping of IP addresses to force elements has occurred in the operational environment. The system allows organic personnel to make changes and update the IP addresses.
- Scenario 18: A unit is overrun during operations. The system detects the unauthorized users and generates a security alert. An authorized person can "zeroize" the system and shut it down remotely.
- Scenario 57: A WIN-T gateway node has been corrupted due to a network intrusion in an operational environment. The intrusion and corruption is detected and another node takes over the corrupted nodes' duties.
- Scenario 22: Following a complete replan during peak operational traffic, system does not allow management traffic to intrude on the operational bandwidth.
- Scenario 58: A legacy application replaces an organic service with a WIN-T service.
- Scenario 28: JNMS is hosted in NOSC-Y, and the army is the combatant commander. The JNMS software runs.
- Scenario 59: A software problem is detected that affects all nodes in an operational environment. The problem is detected, isolated, and logged. Information is conveyed to maintainers for development of a patch.
- Scenario 32: Application has published some situational awareness data during operations, and the data is available within the required time.
- Scenario 36: Ops Tempo changes. Proper policies for new Ops Tempo are executed within 5 minutes.

[7] Scenarios #13 and #63 are identical.

The examples of analysis that follow are typical of the kind of analysis that occurs for each scenario generated during an ATAM-based evaluation. These examples illustrate how scenarios feed analysis, which then identifies risks, sensitivity points, and tradeoffs.

4.8 Scenario Analysis

In this section, we highlight two of the analyzed scenarios, to give a flavor of the results.

4.8.1 Scenario #69 Analysis

A collection of vehicles collects in a "parking lot" during operations. The networks can self-organize, and nodes configure themselves within 5 minutes.

This scenario describes the deployment of a force. At some point, the force's vehicles assemble in a parking lot and begin turning on communications equipment. The WIN-T equipment contained within the force then self-organizes and configures the networks based on which equipment is in the parking lot and the configurations defined in the WIN-T plan.

The self-organization of networks has not been fully defined at this point. The algorithms that are going to be used have not been designed completely, so there is no way to fully evaluate the feasibility or performance. An assumption for this scenario is that all the nodes have been provisioned with the current operating plan. This means that among other things, frequencies have been assigned, policies have been defined and validated, and configuration variables have been determined. The scenario begins when vehicles in the parking lot begin powering on their WIN-T equipment, including radios. The equipment goes through power-on initialization and, it is thought, will idle in a state waiting for the operator to input a command to allow it to begin transmitting. Transmission hardware plays the initial role, since it begins transmitting messages on a hailing band to any surrounding nodes. That task is under control of the radio firmware. Nodes become aware of each other through reception of these hailing messages. The nodes receive position, power, and identification information from other nodes and use it to build the networks.

In a practical situation, it is probably not possible to have full interconnectivity, since the number of nodes will be too great. Configuration parameters are used to limit the number of permissible neighbors and to select which known nodes will be connected to. QoS in the system is used to throttle the messages processed. Nodes will have to be able to deal with errors such as other nodes being on the wrong frequency. A cost function and override list are used to select from a large number of close nodes that are too numerous for the node to connect to. Certain nodes may be designated as preferred, so the ad hoc collection of links will be pruned to include the preferred nodes as they become known.

Configuration parameters are managed from NetOps software through policies that get downloaded into radio firmware. There are currently no performance models for predicting performance through the startup of nodes and self-organization of the networks.

Multiple levels of connection establishment are involved. Low levels are mediated via the low power line-of-sight radios. Higher levels involve the routers and higher level nodes that distribute routing information and other data, and eventually make the set of vehicles ready to pass operational and management traffic among users.

Hasty reorganizations will probably require a replan, but the establishment of a new network topology and membership is an algorithmic problem and probably not an architectural one. There appears to be some uncertainty with regard to requirements that make it difficult to evaluate whether the design is sufficient or the approach is overdesigned. In particular, the scale of the problem (such as numbers of nodes, links, networks to be supported, etc.) is a driver of the complexity of the solution.

Interfacing to the radios requires an understanding of the radio Management Information Base (MIB), both for operation and for modeling the data. The architecture is apparently knowledgeable of the format of this MIB, so it is not schema neutral. Simple changes to outside systems, such as data changes in the MIB, may induce changes in WIN-T as a consequence.

Risks:

- Since algorithms have not been fully defined, the feasibility and performance of the design cannot be fully evaluated.
- There are no performance models to determine the performance envelope of the design.
- Uncertain requirements make it difficult to determine whether the design is necessary and sufficient or more than sufficient.
- The architecture is not schema neutral, so new schemas that must be accommodated have a cost impact on the architecture. (However, this risk is somewhat peripheral to the scenario.)

Non-Risks:

- There is a high degree of flexibility through the configuration parameters.
- The solution to building the network is algorithmic and flexible to any potential network that might be built.

Sensitivity Points:

No sensitivity points were captured.

Tradeoff Points:

No tradeoff points were captured.

4.8.2 Scenario #18 Analysis

A unit is overrun during operations. The system detects the unauthorized users; generates a security alert; an authorized person can "zeroize" the system and shut it down remotely.

This scenario illustrates the situation where a WIN-T node that is operating within a WIN-T network environment is captured intact by an enemy. The situation is that the WIN-T operators were unable to disable the equipment, and it was captured fully connected and operational with the enemy able to access the system and perform operations at a WIN-T terminal device. The system can monitor actions by the unauthorized user and provide an alert to other nodes that unauthorized actions are being attempted. An operator at a remote location is notified of the situation and can evaluate it and make a decision to disable the captured node from a remote location.

There are two aspects to this scenario: (1) the detection of the unauthorized intrusion and (2) the action taken to rectify the situation. Detecting an unauthorized intrusion automatically is not a trivial task. If the intruder attempts to access areas that the user currently logged on is not authorized to access or performs other actions atypical of the user, an intrusion may be detected by algorithmic processing. The reliability of this sort of solution depends on being able to unambiguously characterize unauthorized or unusual activity. In many cases, such a characterization may not be possible, so detection may ultimately have to fall to human operators at other nodes. Even in the face of automatic detection, it may still be desirable for the system to alert a human operator of suspicious activity and rely on the human to make the final determination of whether the access is unauthorized.

Correction of the situation involves some form of disabling the compromised node. Doing so would probably require, as a minimum, the destruction of any disk drives on the node and the "zeroization" of memory and any keys. More destructive means are available but somewhat problematic when employed by anyone not physically located at the node equipment. It is likely that any form of destruction will require human intervention, whether remote or local. A number of threads can be postulated including remote "zeroize," local, operator-initiated destruction, the system sending an alarm message and requesting some other node to resolve the problem, and so forth.

The WIN-T architecture provides component types (service components) that can be instantiated with algorithmic means to automatically detect intrusions and remotely trigger the disabling of another node. These services could make use of the rules engine that is part of WIN-T. However, there are many doctrinal, policy, and safety issues that impact the desirability or capability to carry out remote destruction. The architecture does not inhibit the introduction of these capabilities, but their feasibility is an algorithmic or heuristic problem, not an architectural one. It may be useful to include support for "zeroization" as a node state should the capability be included in WIN-T. State machines are often more amenable to managing fundamental capabilities across an entire system.

Risks:

- The requirements for the scenario are not worked out. There are doctrinal issues as well as safety and security issues.
- There is no currently identified service or component that has been allocated the responsibility for destruction.
- Services must be able to authenticate logins or access from other locations; that feature has not been included yet.

Non-Risks:

No non-risks were captured.

Sensitivity Points:

No sensitivity points were captured.

Tradeoff Points:

No tradeoff points were captured.

4.9 Summary and Risk Themes

Fifteen scenarios were analyzed during the WIN-T evaluation. From these scenarios, 25 risks and 9 non-risks were identified. A smaller number of sensitivity and tradeoff points emerged. These are typical numbers. Although the evaluation team makes every effort to document non-risks, sensitivity points, and tradeoff points, limitations often compel the team to give highest priority to capturing risks.

From the compiled list of all discovered risks, the evaluation team synthesizes a set of risk themes—themes that seem to be common among several of the risks and may represent areas of systemic or large-scale exposure to risk in ways that threaten the business drivers for the system. In the case of WIN-T, three themes emerged:

1. uncertainty in requirements. This theme had to do with a number of areas where requirements are not yet tied down, compelling the architects to make guesses that are educated but may result in a large amount of rework in the future.
2. lack of documentation or specificity regarding technologies, products, evolving standards, and interfaces. This theme dealt with the reliance on (for example) standards that are only now emerging and are not yet concretely defined and adopted.
3. Insufficient models for predicting resource requirements. This theme dealt with the observed inability to make assertions about, among other things, whether the architecture will be able to meet its performance goals.

The ramifications are

- Requirements uncertainties (risk theme #1) and information deficiencies (risk theme #2) make it difficult to know what to build, to meet cost and schedule goals, and to interoperate with other systems that WIN-T needs to accommodate.
- Insufficient models (risk theme #3) risk the possibility that the system will not meet KPPs, especially those related to performance. It is also difficult to understand whether the hardware environments are sufficient to run the software.

4.10 Final Presentation

After analysis is complete, the evaluation team caucuses for an hour or so and prepares a viewgraph presentation recapping the evaluation and presenting conclusions. The business drivers and architecture are summarized, and the utility tree and brainstormed scenarios are revisited. A synopsis of each scenario analysis is presented, followed by a compendium of the risks, non-risks, sensitivity points, and tradeoff points discovered. Finally, the risk themes are presented, along with the ramifications each holds for the business drivers, if not addressed.

The final presentation for WIN-T took approximately 90 minutes and included the information described above.

5 Post-ATAM Activities

The most immediate benefit of conducting the ATAM-based architecture evaluation was increased communication between the government stakeholders and software developers. WIN-T also benefited significantly from the communication opportunities between the participating contractors who are now jointly collaborating on the development effort as a result of the aforementioned recent merger. One of the development contractors is located in Massachusetts and the other in Maryland. While the contractors' software leads routinely meet with the members of their counterpart teams, this evaluation gave other members of the contractor software development teams an opportunity to meet some of their counterparts. And it also gave the systems engineering personnel an opportunity to meet with their counterparts and discuss the architectural decisions.

At the end of the ATAM evaluation, participants were asked to complete an evaluation form. Rather than just checking the boxes in order to be done quickly, every single participant chose to report positive comments such as

- *"Thought provoking. Undiscovered relationships revealed. Believe similar process should be performed at requirements generation stage in coordination with TRADOC."*
- *"The ATAM proved very beneficial in documenting the good decisions made to date."*
- *"The interfaces being ill-defined was a surprise to me."*
- *"Evaluation was worthwhile due to communication between the design team and the customer and communication between the system and software design teams."*
- *"Identified and clarified issues such as life-cycle maintenance."*
- *"I believe that this was done very well from the standpoint that folks weren't inhibited, defensive, or unreasonable—all input was considered and appreciated."*
- *"Our design decisions will be more user-scenario based."*
- *"Extremely useful and should become a part of all programs."*
- *"Identified many of the risks that we need to mitigate."*

One of the first efforts initiated after the ATAM evaluation was to revise the software architecture documentation. Because of the merger of the developing contractors, the architecture documentation had inconsistencies and was excessively complex. Often, a figure that was developed by one partner before the merger was adopted and modified by the other partner and then added to by both partners. As a result, diagrams were sometimes complex, confusing, and error prone.

As a result of the input from the ATAM evaluation and follow-on work by the WIN-T developers, an updated software architecture document has been delivered to the government.

The architecture documentation is greatly improved. For example, the layered view diagram shown in Figure 3 on page 18 had inconsistencies and errors as a result of its mixed heritage. It was redone as shown in Figure 10 and is subsequently being used as the basis for many other diagrams in the software architecture description and other WIN-T documents.

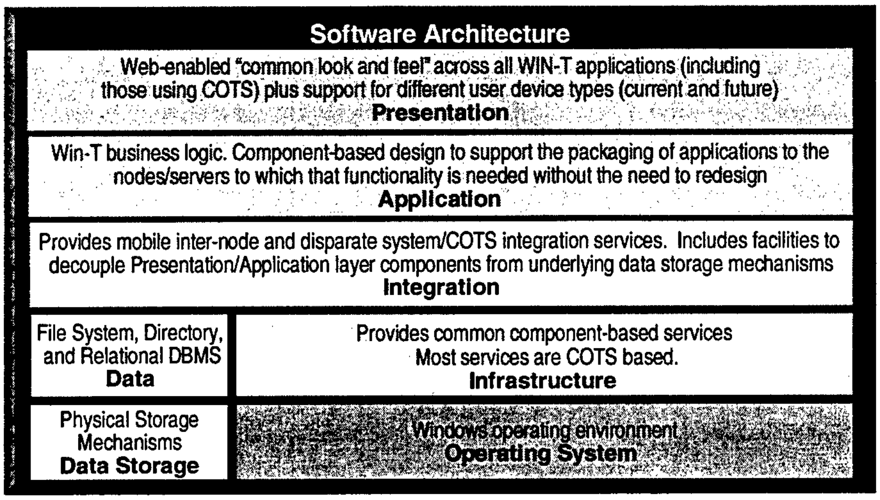


Figure 10: Software Architecture Layered Pattern

Figure 11 shows an example of this improved layer diagram being used as a basis for other diagrams. That diagram provides details about services being provided by various CSCIs within this layer pattern.

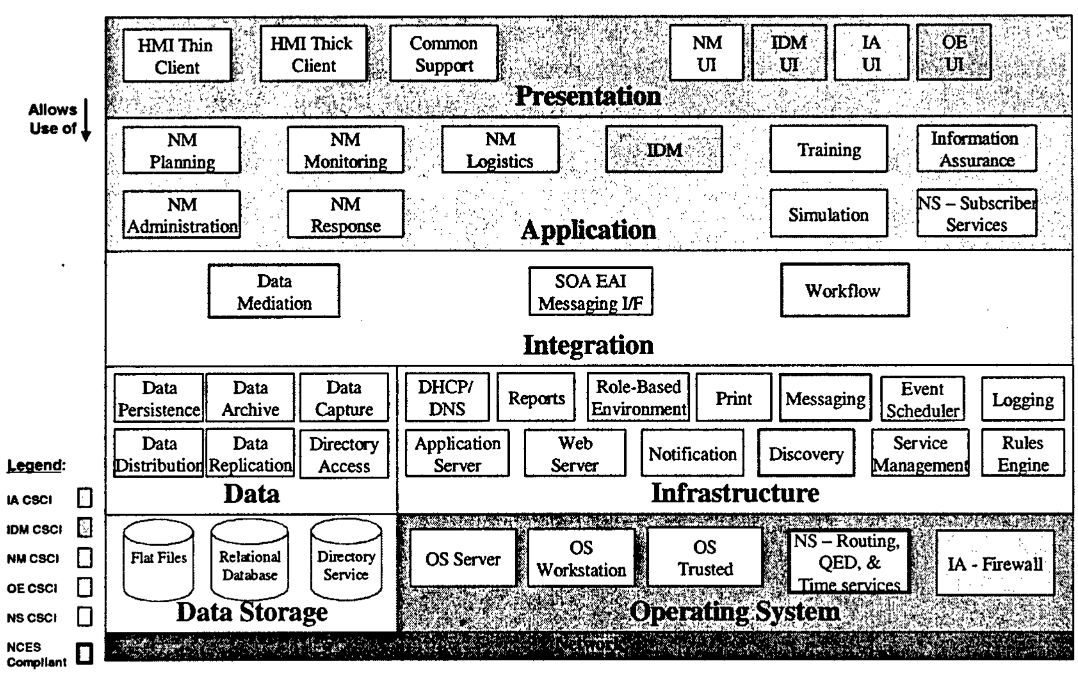


Figure 11: Software Architecture Layered Pattern (Details)

The decomposition view shown in Figure 6 on page 21 is another example of the improvements in the documentation. Originally, it also had problems with mixed origins and was revised as shown in Figure 12. This revised diagram is also now used in other documents such as the Life-Cycle Cost Estimate.

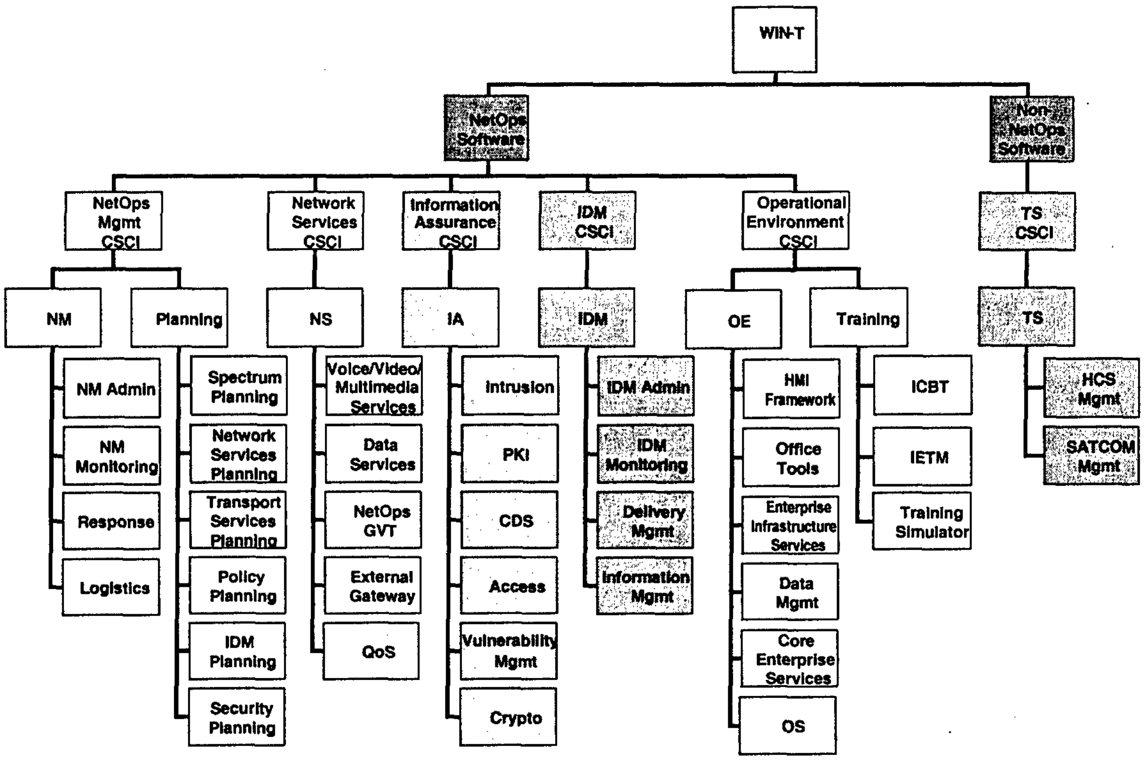


Figure 12: Functional Decomposition

Software risk management is another area of improvement within the program after the ATAM. Several new risks related to software have been, or are being, added to the program risk management database, along with mitigation and contingency plans. Risks, and their mitigation steps, are actively monitored, managed, and briefed.

The first risk to be entered and tracked is the possibility of not meeting the schedule for the selection (development) and integration of COTS, government off-the-shelf (GOTS), and developed software applications that meet the program requirements. Not meeting the schedule would result in the failure to meet the program schedule. The mitigation involves progress monitoring and decision points regarding the selection or development of alternate software applications to be integrated.

The second major risk is the possible failure to integrate the processes and standards of two different companies, working as partners, into a well-integrated team. Such a failure could result in missed schedules, software that fails to perform due to interoperability issues, and software that will be more difficult and expensive to maintain due to a lack of standards. The mitigation is the development of a Combined Team Software Development Plan, covering

processes and standards, and a Combined Team Capability Maturity Model® Integration (CMMI®) evaluation.

Another major risk is the possibility of failing to meet the Army, Joint, and coalition interoperability requirements due to evolving and changing interfaces and standards. These systems are being developed at the same time WIN-T is being developed, resulting in a moving target for software development. The possible result is a system that fails to meet the interoperability performance requirements or doesn't have the resources and time needed to rework the software to meet those requirements.

Additional risks will likely surface as additional scenarios are analyzed.

Finally, a software IPT has been chartered at the request of the PD with membership similar to the ATAM stakeholders. This IPT has been charged with analyzing the remaining scenarios, developing additional scenarios as required, and monitoring the various stakeholders' interests.

Based on the results and favorable impressions from the WIN-T ATAM, the CECOM SEC is trying to do two things: (1) to schedule additional training in software architecture and the ATAM in order to better understand the importance of a well-thought-out and documented software architecture and (2) to be able to provide ATAM-based architecture evaluations as a service to all the programs in the CECOM community.

® Capability Maturity Model and CMMI are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

6 Conclusion

In this technical note, we have discussed applying the ATAM during the development of a large government-sponsored tactical communications system. The note presents a general overview of the ATAM process and the results of this ATAM-based evaluation. It also presents benefits that both the acquirer and developer received.

A post-evaluation survey of the participants showed that the WIN-T ATAM-based evaluation was considered a success. A number of specific benefits were reported:

- The ATAM evaluation provided a good opportunity for communication between
  - software developers and stakeholders
  - software developers and systems developers
  - partner developers of the new Combined Team
  - different groups of stakeholders
- The ATAM evaluation highlighted and clarified several previously untracked risks so that they could be monitored and mitigated to reduce their likelihood and impact.
- The ATAM helped the stakeholders understand the nature and importance of the software development effort. On an integration effort, the process is generally viewed as little more than selecting products and writing some "glue ware." Building a brick house that is robust and meets the owner's expectations is more than selecting the bricks and sticking them together with mortar.

Lessons applicable to ATAM evaluations in general were also uncovered:

- An ATAM evaluation can be applied successfully in a government-owned contractor-operated environment. Two important factors leading to success were (1) the flexibility of the existing task-order contract and (2) the excellent relationship that existed between the government and the contractors.
- Even though there is typically not enough time to analyze all the scenarios during a two-day ATAM evaluation, it is possible for the participants to continue the analysis without the coaching of the ATAM evaluation team.

CECOM and the SEI have had a long-standing strategic collaboration to apply emerging software technologies. CECOM provides an excellent example of how a government organization can incorporate these technologies to solve real problems and improve its mission effectiveness.

Appendix A Acronym List

1 CAV	First Cavalry
4 ID	Fourth Infantry Division
ACAT	acquisition category
API	application program interface
ASSIP	Army's Strategic Software Improvement Program
ATAM	Architecture Tradeoff Analysis Method
BCFO	Battlefield Command Futures Office
BPEL	Business Process Execution Language
C3T	Command, Control, and Communications Tactical
CECOM	Communications and Electronics Command
CE LCMC	Communications Electronics Life Cycle Management Center
CERDEC	Communications Electronics Research, Developments, and Engineering Center
COE	common operating environment
CONUS	continental United States
COTS	commercial off-the-shelf
CMMI	Capability Maturity Model Integration
CSCI	computer software configuration item
DBMS	database management system
DoD	Department of Defense
EAC	Echelons Above Corps

EID	Enterprise Information Dashboard
FCS	Future Combat System
GIG	global information grid
GOTS	government off-the-shelf
GUI	graphical user interface
HFE	Human Factors Engineering
HMI	human machine interface
IA	Information Assurance
IDM	Information Dissemination Management
IER	Information Exchange Requirement
IEWS	Intelligence, Electronic Warfare, and Sensors
IOT	Initial Operational Test
IP	Internet protocol
IPT	integrated product team
ISYSCON	Integrated Systems Control
J2EE	Java 2 Enterprise Edition
JMS	Java Message Service
JNMS	Joint Network Management System
JTRS	Joint Tactical Radio System
KO	contracting officer
KPP	key performance parameter
LPI/LPD	Low Probability of Intercept/Low Probability of Detection
LRC	Logistics Readiness Command
MB	Maneuver Brigade
MIB	Management Information Base

MSE	Mobil Subscriber Equipment
MVC	Model View Controller
NCES	Network Centric Enterprise Services
NetCOP	Network Common Operating Picture
NetOps	Network Operations
NM	NetOps Management
NOSC-Y	Network Operations Center-Y
NS	Network Services
OE	operational environment
OE HLD	Operating Environment High-Level Design
OPORD	Operations Order
Ops	operations
ORD	Operational Requirements Document
OS	operating system
PEO	Program Executive Office
PD	project director
PM	program manager
QoS	quality of service
RDECOM	Research, Development, and Engineering Command
S&TCD	Space and Terrestrial Communications Directorate
SEC	Software Engineering Center
SED	Software Engineering Directorate
SEI	Software Engineering Institute
SOA	service-oriented architecture
SOAP	Simple Object Access Protocol

SoS	system of systems
SW	software
SYSCON	Systems Control
TBR	to be resolved
TEP	Task Execution Plan
TIM	Technical Interchange Meeting
TRADOC	Training and Doctrine Command
TRI-TAC	Tri-Service Tactical Communications System
TS	Transmission Subsystem
TSM	TRADOC Systems Manager
UA	Unit of Action
UDDI	Universal Description, Discovery, and Integration
UE	Unit of Employment
UI	user interface
VM	virtual machine
WIN-T	Warfighter Information Network-Tactical
XML	Extensible Markup Language

References

URLs are valid as of the publication date of this document.

[Bass 03] Bass, L.; Clements, P.; & Kazman, R. *Software Architecture in Practice, Second Edition.* Boston, MA: Addison-Wesley, 2003.

[Clements 02a] Clements, P.; Bachmann, F.; Bass, L.; Garlan, D.; Ivers, J.; Little, R.; Nord, R.; & Stafford, J. *Documenting Software Architectures: Views and Beyond.* Boston, MA: Addison-Wesley, 2002.

[Clements 02b] Clements, P.; Kazman, R.; & Klein, M. *Evaluating Software Architectures.* Boston, MA: Addison-Wesley, 2002.

[Clements 96] Clements, P. & Northrop, L. *Software Architecture: An Executive Overview* (CMU/SEI-96-TR-003, ADA305470). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996. http://www.sei.cmu.edu/publications/documents/96.reports /96.tr.003.html.

[Jones 99] Jones, L. G *Product Line Acquisition in the DoD: The Promise, The Challenges* (CMU/SEI-99-TN-011, ADA373184). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. http://www.sei.cmu.edu/publications/documents/99.reports /99tn011/99tn011abstract.html.

[Kazman 00] Kazman, R.; Klein, M.; & Clements, P. *ATAM: Method for Architecture Evaluation* (CMU/SEI-2000-TR-004, ADA382629). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000. http://www.sei.cmu.edu/publications/documents /00.reports/00tr004.html.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.			
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE September 2005		3. REPORT TYPE AND DATES COVERED Final
4. TITLE AND SUBTITLE Using the SEI Architecture Tradeoff Analysis Method to Evaluate WIN-T: A Case Study			5. FUNDING NUMBERS FA8721-05-C-0003
6. AUTHOR(S) Paul Clements, John Bergey, David Mason			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213			8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2005-TN-027
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116			10. SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES			
12A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS			12B DISTRIBUTION CODE
13. ABSTRACT (MAXIMUM 200 WORDS) The software architecture of a software-intensive system greatly determines system quality. Evaluating that architecture for fitness of purpose before the system is implemented or undergoes a major modification is a cost-effective approach to uncovering deficiencies early and reducing risk. When used appropriately, software architecture evaluations can have a favorable effect on a delivered or modified government system. This technical note describes the application of the SEI Architecture Tradeoff Analysis Method® (ATAM®) to the U.S. Army's Warfighter Information Network-Tactical (WIN-T) system. The WIN-T system is being developed by a government-contractor team headquartered at the U.S. Army's Communications and Electronics Command (CECOM) in Ft. Monmouth, New Jersey. This technical note presents the WIN-T program context, the definition of software architecture, and the background of the WIN-T organization and system being evaluated. It also provides a general overview of the ATAM process, describes the application of the ATAM to the WIN-T system, presents important results, and summarizes the benefits the program received.			
14. SUBJECT TERMS case study, ATAM, Architecture Tradeoff Analysis Method, architecture, architecture evaluation, architecture tradeoff, architecture analysis, quality attributes, quality attribute scenario, evaluation method, CECOM, WIN-T			15. NUMBER OF PAGES 66
16. PRICE CODE			
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL